

What is claimed is:

- 1 1. A system comprising:
 - 2 a first module adapted to receive a set of queries and to provide a set of candidate
 - 3 indexes for the set of queries, the first module adapted to eliminate one or more candidate
 - 4 indexes based on one or more predetermined criteria; and
 - 5 an optimizer adapted to generate a recommended index from the set of candidate
 - 6 indexes.
- 1 2. The system of claim 1, wherein the set of queries comprises a set of SQL
- 2 statements.
- 1 3. The system of claim 1, wherein the optimizer is adapted to generate at least
- 2 another recommended index from the set of candidate indexes.
- 1 4. The system of claim 1, wherein the optimizer is adapted to use statistics.
- 1 5. The system of claim 4, wherein the statistics are based on a scan of a sample of
- 2 one or more tables, the sample less than all the rows of the one or more tables.
- 1 6. The system of claim 5, further comprising a user interface to receive an indication
- 2 of a user-specified size of the sample.
- 1 7. The system of claim 1, wherein the one or more predetermined criteria comprises
- 2 a threshold change rate, the first module adapted to eliminate one or more candidate
- 3 indexes having a change rate exceeding the threshold change rate.
- 1 8. The system of claim 7, wherein the first module is adapted to further eliminate a
- 2 candidate index that is a subset of another candidate index.

1 9. The system of claim 1, further comprising an analysis module adapted to apply a
2 genetic algorithm, the analysis module adapted to cooperate with the optimizer to
3 generate the recommended index.

1 10. The system of claim 9, wherein the first module is adapted to provide the set of
2 candidate indexes by identifying the candidate indexes from the set of queries and
3 defining the set of queries in a database.

1 11. The system of claim 10, wherein the analysis module is adapted to access the
2 database to retrieve the candidate indexes.

1 12. The system of claim 10, further comprising a validation module adapted to
2 validate the recommended index in a database system.

1 13. The system of claim 12, further comprising a user interface to receive user-
2 specified one or more indexes, the optimizer adapted to generate a cost associated with a
3 query plan based on the user-specified one or more indexes.

1 14. The system of claim 13, wherein the user interface is adapted to receive a user-
2 specified percentage value, the system further comprising another module to collect
3 statistics based on a sample of rows of one or more tables, a size of the sample based on
4 the user-specified percentage value.

1 15. The system of claim 14, further comprising another module adapted to provide a
2 hint on which table or tables statistics need to be collected.

1 16. The system of claim 10, wherein the analysis module is adapted to access the
2 database to retrieve the candidate indexes.

1 17. The system of claim 1, further comprising an analysis module adapted to apply a
2 predetermined algorithm, the analysis module adapted to cooperate with the optimizer to
3 generate the recommended index.

1 18. The system of claim 17, wherein the analysis module is adapted to submit
2 candidate indexes to the optimizer, the optimizer adapted to determine the cost of one or
3 more of the queries based on the candidate indexes.

1 19. The system of claim 18, wherein the optimizer is adapted to select the candidate
2 index associated with a lowest cost as the recommended index.

1 20. The system of claim 1, wherein the set of queries comprises a workload captured
2 in a database system, and wherein the database system is a parallel system having plural
3 access modules.

1 21. The system of claim 20, wherein the optimizer is adapted to compute costs for the
2 candidate indexes for the database system.

1 22. A method of selecting a recommended index for a database system, comprising:
2 receiving a workload containing a set of queries of the database system;
3 generating a set of candidate indexes from the workload;
4 removing candidate indexes based on one or more predetermined criteria; and
5 invoking an optimizer to provide cost analysis to generate the recommended
6 index from the set of candidate indexes.

1 23. The method of claim 22, further comprising applying a predetermined algorithm
2 to identify the recommended index.

1 24. The method of claim 23, wherein applying the predetermined algorithm
2 comprises applying a genetic algorithm.

1 25. The method of claim 24, further comprising selecting the candidate index having
2 a lowest cost as the recommended index.

1 26. The method of claim 22, further comprising providing graphical user interface
2 screens to receive user input for selecting the recommended index.

1 27. The method of claim 26, wherein providing graphical user interface screens
2 comprises displaying an activatable item to perform workload identification to identify
3 the workload.

1 28. The method of claim 27, wherein providing the graphical user interface screens
2 further comprises displaying another activatable item to perform workload definition to
3 save the workload into a database.

1 29. The method of claim 28, wherein providing the graphical user interface screens
2 further comprises displaying another activatable item to perform index analysis to
3 analyze the candidate indexes to generate the recommended index.

1 30. The method of claim 29, wherein providing the graphical user interface screens
2 further comprises displaying another activatable item to validate the recommended index
3 in the database system.

1 31. The method of claim 29, wherein providing the graphical user interface screens
2 comprises displaying another activatable item to validate the recommended index in a test
3 system having an emulated environment of the database system

1 32. The method of claim 30, wherein providing the graphical user interface screens
2 further comprises displaying another activatable item to cause submission of a command
3 to the database system to create the recommended index.

1 33. The method of claim 26, wherein providing the graphical user interface screens
2 comprises displaying one or more reports relating to the recommended index.

1 34. The method of claim 33, wherein providing the graphical user interface screens
2 further comprises displaying a comparison of a cost using the recommended index with a
3 cost using an existing index.

1 35. The method of claim 33, wherein providing the graphical user interface screens
2 further comprises displaying cost improvement relating to use of the recommended index.

1 36. The method of claim 22, wherein invoking the optimizer is performed in a test
2 system separate from the database system.

1 37. The method of claim 36, further comprising importing environment information
2 of the database system into the test system to emulate the database system in the test
3 system.

1 38. The method of claim 37, wherein importing the environment information
2 comprises importing the environment information of a parallel database system having
3 plural access modules.

1 39. The method of claim 22, wherein invoking the optimizer is performed in the
2 database system.

1 40. An article comprising at least one storage medium containing instructions that
2 when executed cause a system to:
3 receive a set of queries;
4 generate a set of candidate indexes from the set of queries;
5 eliminate candidate indexes based on one or more predetermined criteria;
6 invoke an optimizer to perform cost analysis of the candidate indexes; and
7 use the cost analysis to select a recommended index for a database system.

1 41. The article of claim 40, wherein the instructions when executed cause the system
2 to eliminate a candidate index that has a change rate greater than a preset threshold.

1 42. The article of claim 41, wherein the instructions when executed cause the system
2 to eliminate a candidate index that is a subset of another candidate index.

1 43. The article of claim 40, wherein the instructions when executed cause the system
2 to apply a genetic algorithm to select the recommended index.